

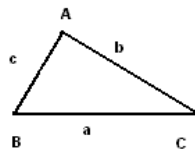
Computing for Scientists - Lab 3

CS 1340 — Dr. Mihail
Department of Computer Science
Valdosta State University

September 6, 2018

1 Introduction

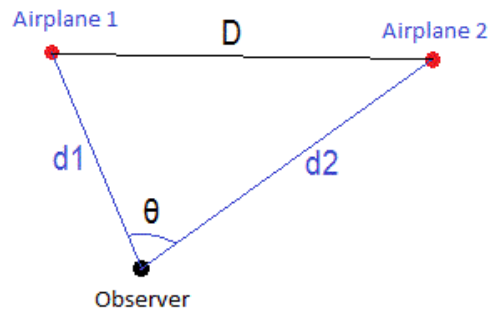
Imagine you work for an ATC (air traffic control) tower, located at point O (observer, see figure below). The radar system detects the distance from O to each of the two aircrafts - $d1$ and $d2$, as well as the angle θ . You are asked to determine distance D , given $d1$, $d2$ and θ . The problem is an application of the Cosine Law, which in its general case can be stated as:



$$a^2 = b^2 + c^2 - 2 * b * c * \cos(\theta) \quad (1)$$

In our particular case, D can be computed as follows:

$$D^2 = d1^2 + d2^2 - 2 * d1 * d2 * \cos(\theta) \quad (2)$$



Sample Runs Your program needs to **follow exactly the output format, as in the following sample runs. This includes whitespaces, newlines and number of significant digits (4).**

Sample run 1 (sol.m)

```
>> sol
```

```
****ATC computation****
```

```
Enter d1 (distance 1 from the observer): 40
```

```
Enter d2 (distance 2 from the observer): 72
```

```
Enter angle theta between the two airplanes: 34
```

```
The distance between the two airplanes is: 44.8190
```

Sample run 2:

```
>> sol
```

```
****ATC computation****
```

```
Enter d1 (distance 1 from the observer): 10.23
```

```
Enter d2 (distance 2 from the observer): 10.78
```

```
Enter angle theta between the two airplanes: 3.33333
```

```
The distance between the two airplanes is: 0.8220
```

Sample run 3:

```
>> sol
```

```
****ATC computation****
```

```
Enter d1 (distance 1 from the observer): 72
```

```
Enter d2 (distance 2 from the observer): 50
```

```
Enter angle theta between the two airplanes: 45
```

```
The distance between the two airplanes is: 50.9199
```

2 Design

Decide on what steps you will need to perform to solve this problem. Make a numbered list and save this file as design.txt (in Notepad or similar text editor). For example:

1. Display introductory message
2. Rest of your design steps
3. ...

Submit this file with your additions using the electronic assignment submission page. Choose the menu choices of "Design" and "Program 1".

3 Test Cases

First, read the assignment carefully. Look for how the program is supposed to behave. You do not know what the code looks like - that is fine. This assignment gives some examples of normal runs. There are other test cases needed. Save this doc file: <http://mypages.valdosta.edu/rpmihail/teaching/S15/CS1340/programs/p1/p1testcases.doc> and fill in the table with test cases. Put your name and the section at the top. Submit this file with your additions using the electronic assignment submission page. Choose the menu choices of "TestCases" and "Program 1".

4 Implementing the Design

Write a MATLAB script to implement your design. Follow the design you just created and insert the steps as comments prior to the commands. Make sure you eliminate any syntax and semantics errors. Here is where test cases come in handy! Submit your individual source code with the course assignment page. Choose the menu choices of "Code" and "Lab 1". There are several specifications about how your program should be written.

- This lab uses input; you will have to prompt the user for d1, d2 and theta.
- Make sure you format the lines of the output as described. The line breaks and the punctuation should be as shown. The output messages should be exactly as given.

5 Grading Rubric

Program Correctness (60 points)

Introductory message
Calculations are correct
Calculations are done in assignment statements
(not in output statements)
Displays results correctly, both values
text and formatting (line breaks, etc.)

Testing (15 points)

Program Design (15 points)

Style and documentation (10 points)

Prolog (name & purpose)
Comments
Whitespaces

Due dates

1. Design: due Friday, September 7th, before midnight.
2. Test cases: due Friday, September 7th, before midnight.

3. Code: due Friday, September 7th, before midnight.